Lab 7

Zara Waheed

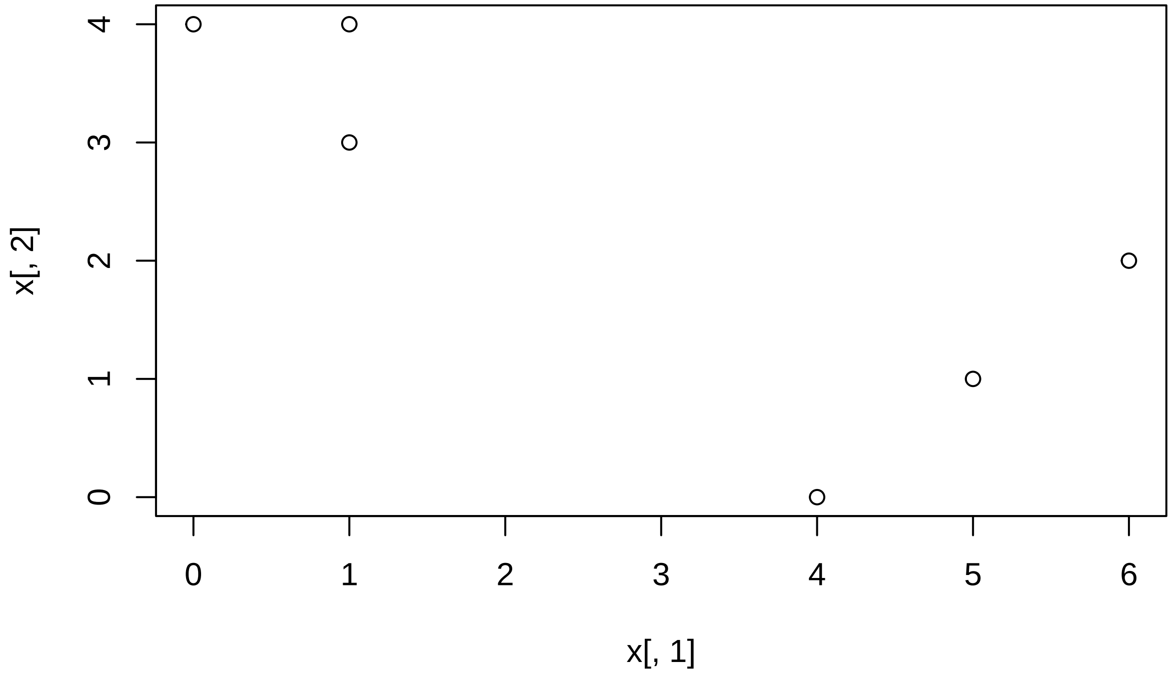
March 31, 2022

Question 12.3

a)

```r
set.seed(100)
x = cbind(c(1, 1, 0, 5, 6, 4), c(4, 3, 4, 1, 2, 0))

plot(x[,1], x[,2])
```



b)

```r
labels = sample(2, nrow(x), replace=T)
labels
```

```
## [1] 2 1 2 2 1 1
```

c)

```r
centroid1 = c(mean(x[labels==1, 1]), mean(x[labels==1, 2]))
centroid2 = c(mean(x[labels==2, 1]), mean(x[labels==2, 2]))
```

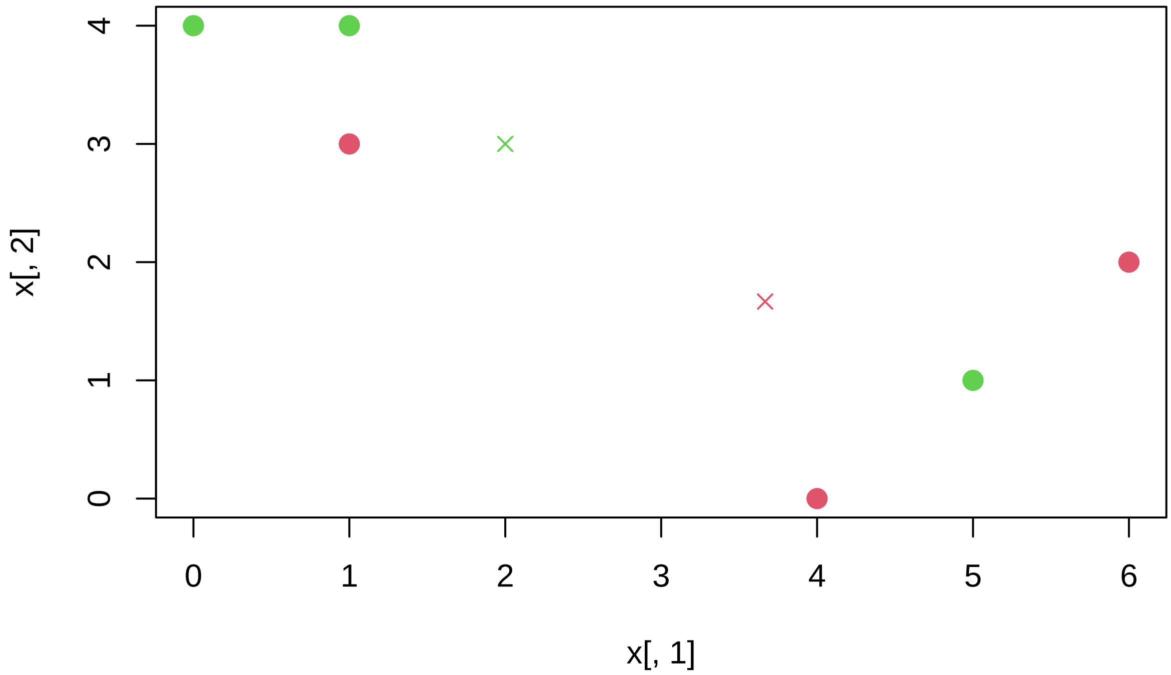
```
centroid1
```

```
## [1] 3.666667 1.666667
```

```
centroid2
```

```
## [1] 2 3
```

```
plot(x[,1], x[,2], col=(labels+1), pch=20, cex=2)
points(centroid1[1], centroid1[2], col=2, pch=4)
points(centroid2[1], centroid2[2], col=3, pch=4)
```



d)

```
euclid = function(a, b) {
  return(sqrt((a[1] - b[1])^2 + (a[2]-b[2])^2))
}
assign_labels = function(x, centroid1, centroid2) {
  labels = rep(NA, nrow(x))
  for (i in 1:nrow(x)) {
    if (euclid(x[i,], centroid1) < euclid(x[i,], centroid2)) {
      labels[i] = 1
    } else {
      labels[i] = 2
    }
  }
  return(labels)
}
labels = assign_labels(x, centroid1, centroid2)
labels
```

```
## [1] 2 2 2 1 1 1
```

e)

```
last_labels = rep(-1, 6)
while (!all(last_labels == labels)) {
  last_labels = labels
  centroid1 = c(mean(x[labels==1, 1]), mean(x[labels==1, 2]))
  centroid2 = c(mean(x[labels==2, 1]), mean(x[labels==2, 2]))
  print(centroid1)
  print(centroid2)
  labels = assign_labels(x, centroid1, centroid2)
}
```

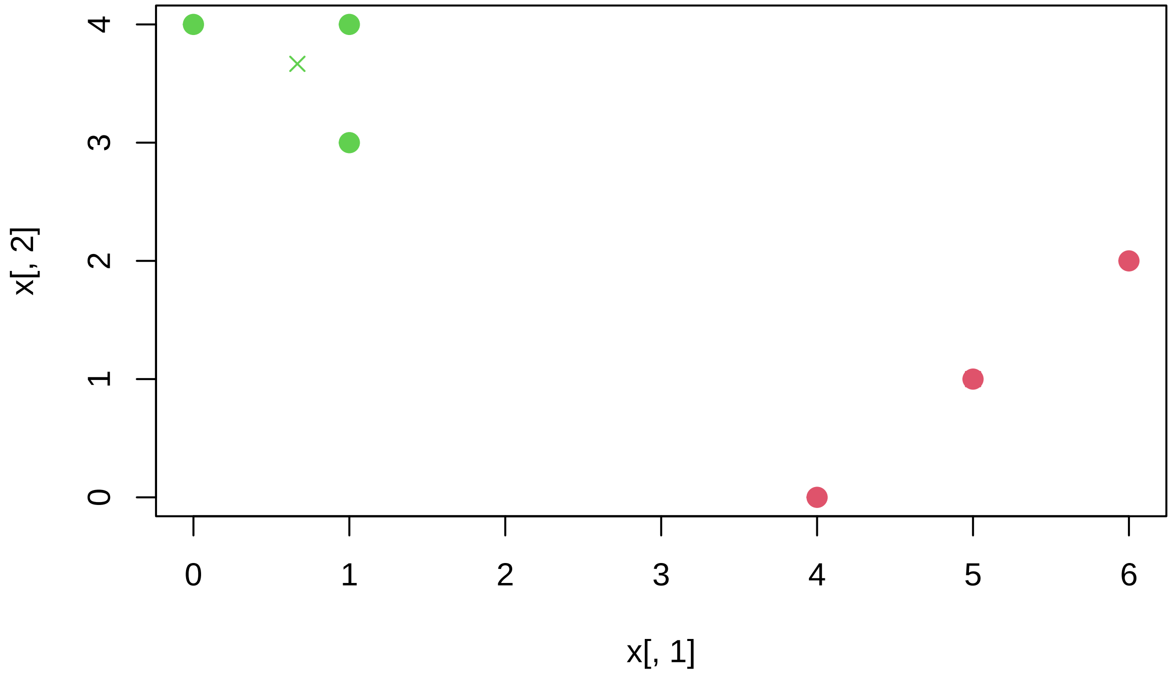
```
## [1] 5 1
## [1] 0.6666667 3.6666667
```

```
labels
```

```
## [1] 2 2 2 1 1 1
```

f)

```
plot(x[,1], x[,2], col=(labels+1), pch=20, cex=2)
points(centroid1[1], centroid1[2], col=2, pch=4)
points(centroid2[1], centroid2[2], col=3, pch=4)
```



Question 12.5

1)

Least socks and computers (3, 4, 6, 8) versus more socks and computers (1, 2, 7, 8).

2)

Purchased computer (5, 6, 7, 8) versus no computer purchase (1, 2, 3, 4). The distance on the computer dimension is greater than the distance on the socks dimension.

3)

Purchased computer (5, 6, 7, 8) versus no computer purchase (1, 2, 3, 4).

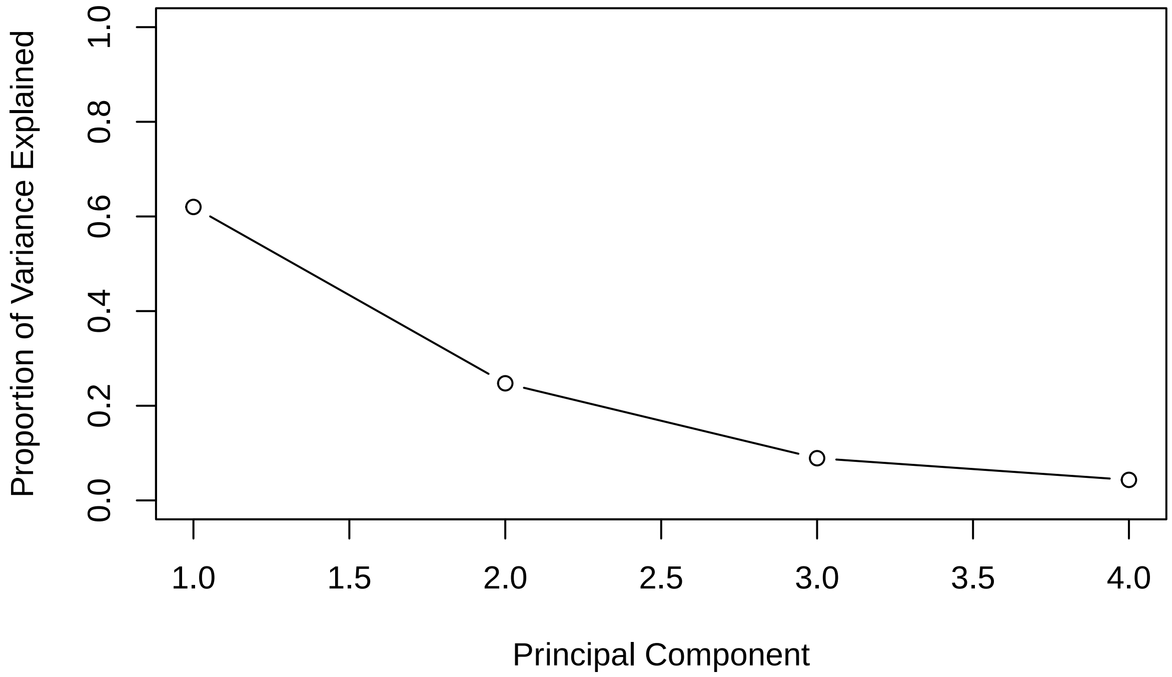
Question 12.8

a)

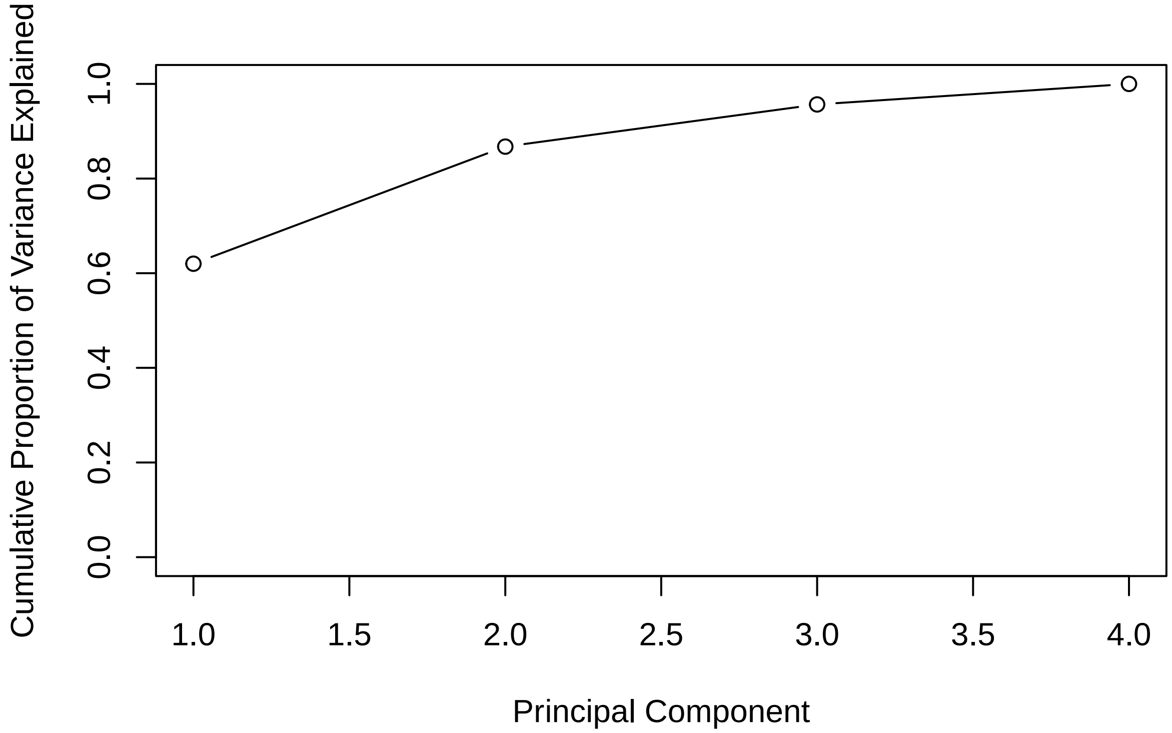
```
rm(list=ls())
attach(USArrests)

pr.out <- prcomp(USArrests,scale=TRUE)
pr.var <- pr.out$sdev^2
pve <- pr.var / sum(pr.var)

plot(pve, xlab="Principal Component", ylab=" Proportion of Variance Explained ",ylim=c(0,1) ,type='b')
```



```
plot(cumsum(pve), xlab = "Principal Component",
ylab = "Cumulative Proportion of Variance Explained", ylim = c(0, 1), type = "b")
```



b)

```
loadings<-pr.out$rotation
USArrests2 <- scale(USArrests)

sum <-sum(as.matrix(USArrests2)^2)
num <-(as.matrix(USArrests2)%*%loadings)^2

col <-c()
for (i in 1:length(num[1,])){
  col[i]<-sum(num[,i])
}

pve1 <- col/sum
pve1
```

```
## [1] 0.62006039 0.24744129 0.08914080 0.04335752
```

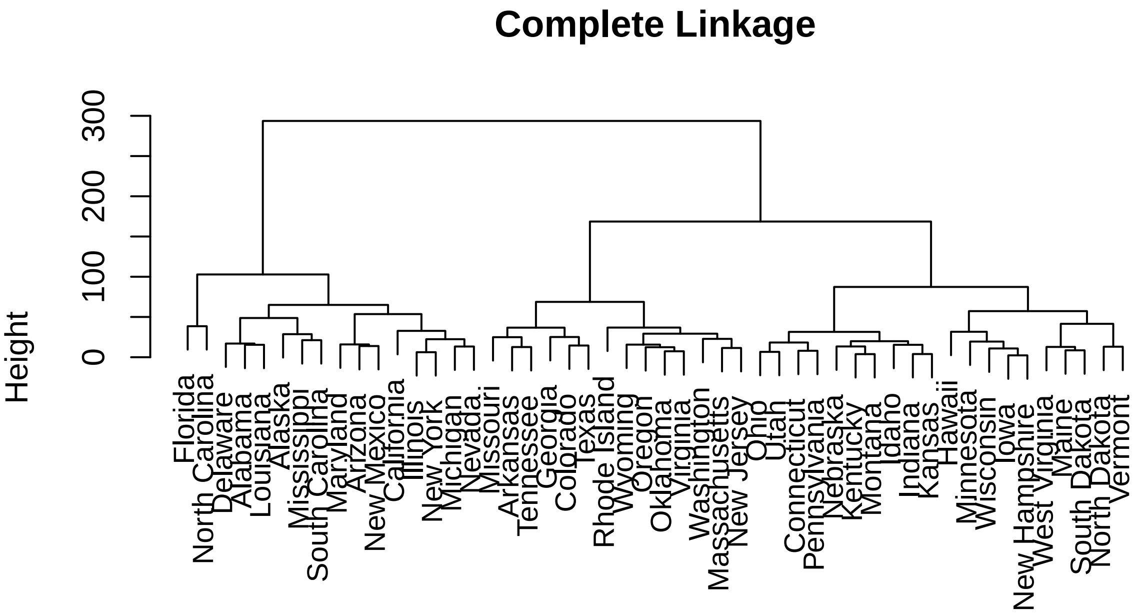
Question 12.9

a)

```
library(ISLR)
arrests = USArrests
hc = hclust(dist(arrests), method = "complete")
```

b)

```
plot(hc, main = "Complete Linkage", xlab = "", sub = "", cex = 0.9)
```



```
hc_3_clust = cutree(hc, 3)

# To see clearly
sort(hc_3_clust)
```

```
##        Alabama         Alaska        Arizona     California       Delaware
##              1              1              1              1              1
##        Florida       Illinois      Louisiana       Maryland       Michigan
##              1              1              1              1              1
##    Mississippi         Nevada     New Mexico       New York North Carolina
##              1              1              1              1              1
## South Carolina       Arkansas       Colorado        Georgia  Massachusetts
##              1              2              2              2              2
##       Missouri     New Jersey       Oklahoma         Oregon   Rhode Island
##              2              2              2              2              2
##      Tennessee          Texas       Virginia     Washington        Wyoming
##              2              2              2              2              2
##    Connecticut         Hawaii          Idaho        Indiana           Iowa
##              3              3              3              3              3
##         Kansas       Kentucky          Maine      Minnesota        Montana
##              3              3              3              3              3
##       Nebraska  New Hampshire   North Dakota           Ohio   Pennsylvania
##              3              3              3              3              3
##   South Dakota           Utah        Vermont  West Virginia      Wisconsin
##              3              3              3              3              3
```

c)

```
# Scale the data
scaled_arrests = scale(arrests)

apply(scaled_arrests, 2, mean)
```

```
##        Murder       Assault      UrbanPop          Rape
## -7.663087e-17  1.112408e-16 -4.332808e-16  8.942391e-17
```

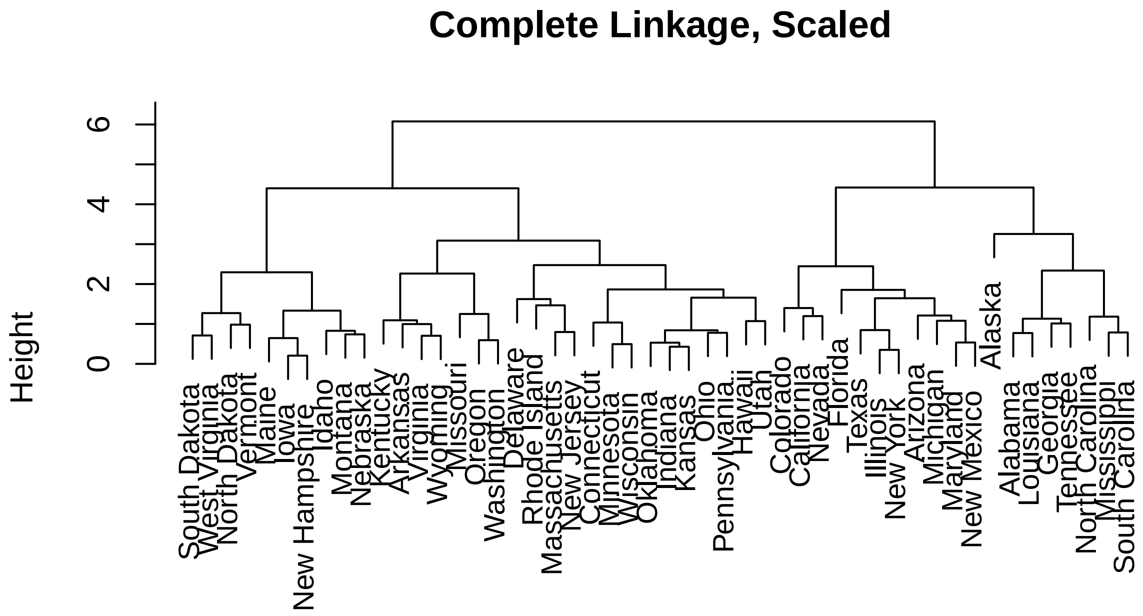
```r
apply(scaled_arrests, 2, var)
```

```
##   Murder  Assault UrbanPop     Rape 
##        1        1        1        1
```

```r
# Cluster
scaled_hc = hclust(dist(scaled_arrests), method = "complete")
```

d)

```r
plot(scaled_hc, main = "Complete Linkage, Scaled", xlab = "", sub = "", cex = 0.9)
```



The variables should be scaled before the inter-observation because the tree looks better after scaling

Question 12.10

a)

```r
x = matrix(rnorm(20*3*50, mean=0, sd=0.001), ncol=50)
x[1:20, 2] = 1
x[21:40, 1] = 2
x[21:40, 2] = 2
x[41:60, 1] = 1
```

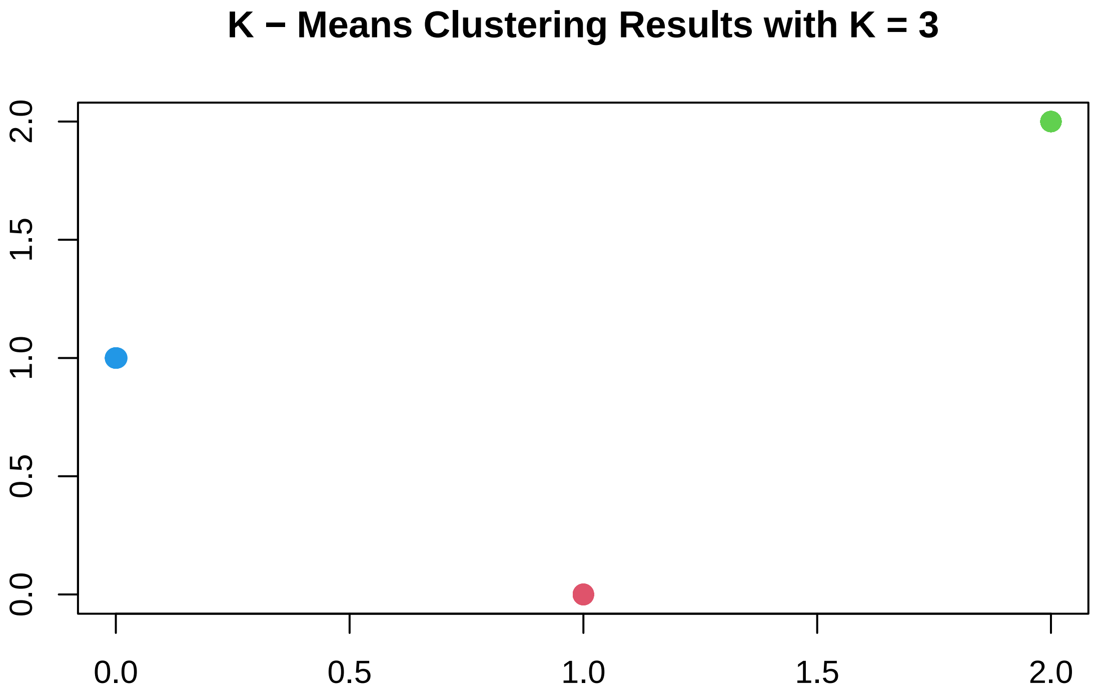
b)

```r
pca_unscaled = prcomp(x, scale = FALSE)
pca_scaled = prcomp(x, scale = TRUE)
```

c)

```r
k = 3
kmeans3 = kmeans(x, k, nstart = 20)
```

```
plot(x,
     col = (kmeans3$cluster + 1),
     main = paste0("K - Means Clustering Results with K = ", k),
     xlab = "",
     ylab = "",
     pch = 20,
     cex = 2)
```



```
table(kmeans3$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20)))
```

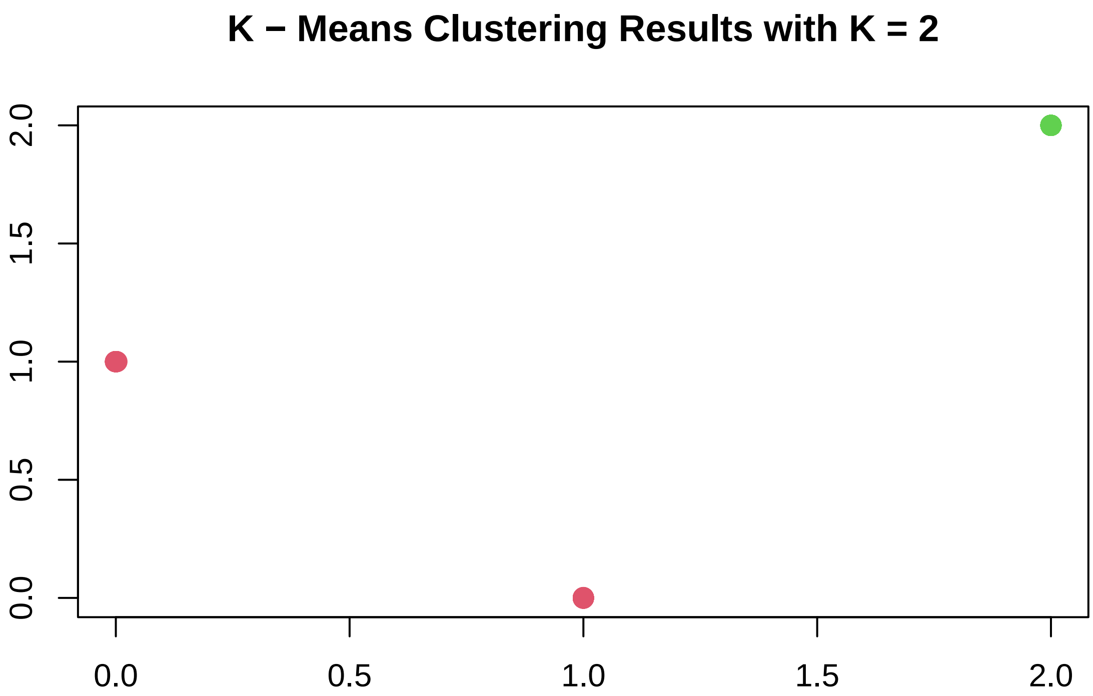
```
## 
##      1  2  3
##   1  0  0 20
##   2  0 20  0
##   3 20  0  0
```

```
sort(kmeans3$cluster)
```

```
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [39] 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
```

d)

```
k = 2
kmeans2 = kmeans(x, k, nstart = 20)
plot(x,
     col = (kmeans2$cluster + 1),
     main = paste0("K - Means Clustering Results with K = ", k),
     xlab = "",
     ylab = "",
     pch = 20,
     cex = 2)
```



```
table(kmeans2$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20)))
```

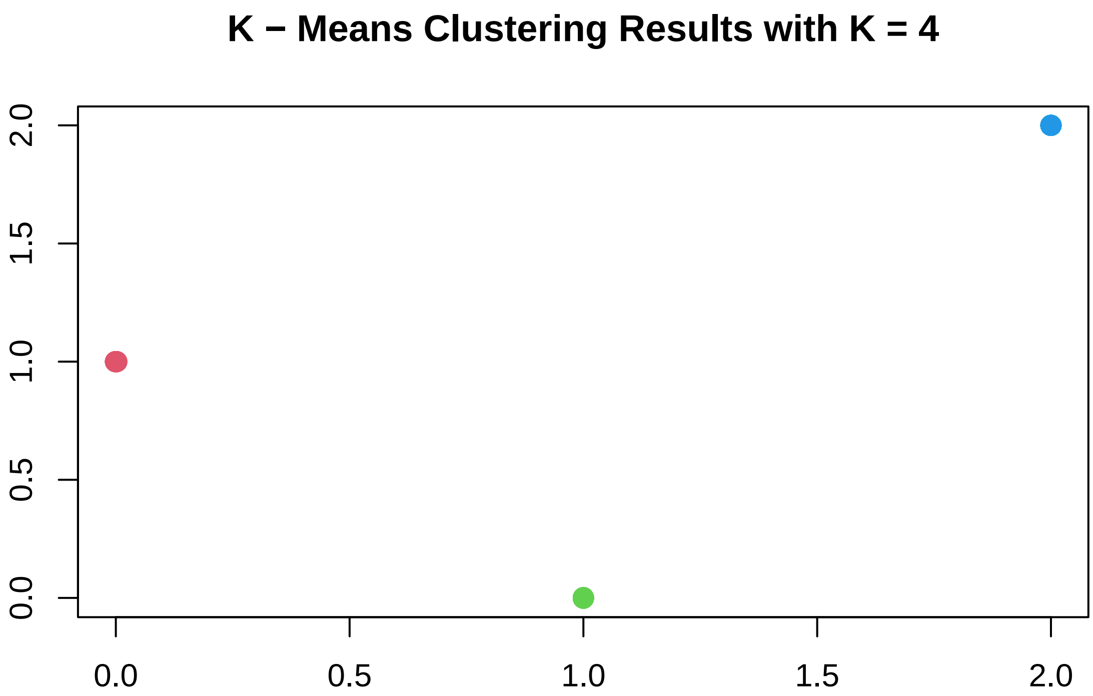
```
##
##       1  2  3
##    1 20  0 20
##    2  0 20  0
```

```
sort(kmeans2$cluster)
```

```
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
## [39] 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
```

e)

```
k = 4
kmeans4 = kmeans(x, k, nstart = 20)
plot(x,
     col = (kmeans4$cluster + 1),
     main = paste0("K - Means Clustering Results with K = ", k),
     xlab = "",
     ylab = "",
     pch = 20,
     cex = 2)
```



```
table(kmeans4$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20)))
```

```
##
##       1  2  3
##    1 20  0  0
##    2  0  0 20
##    3  0 11  0
##    4  0  9  0
```

```
sort(kmeans4$cluster)
```

```
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [39] 2 2 3 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4
```

f)

```
kmeans3_pca = kmeans(pca_unscaled$x[ ,1:2], 3, nstart=20)
table(kmeans3_pca$cluster, c(rep(1, 20), rep(2, 20), rep(3, 20)))
```

```
##
##       1  2  3
##    1  0 20  0
##    2 20  0  0
##    3  0  0 20
```

```
sort(kmeans3_pca$cluster)
```

```
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [39] 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
```

Just like the first c – a perfect match.

g)

```
kmeans3_s = kmeans(scale(x), 3, nstart=20)
sort(kmeans3_s$cluster)
```

```
##  [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [39] 2 2 2 2 2 2 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
```